IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of: Jay H. Connelly	Examiner: Barqadle, Yasin M.
Serial No. 09/533,048	Art Unit: 2153
Filed: March 22, 2000	
For: SIGNALING METHOD AND APPARATUS TO PROVIDE CONTENT ON DEMAND IN A BROADCAST SYSTEM	

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

DECLARATION PURSUANT TO 37 C.F.R.§1.131

Sir:

I, Jay H. Connelly, hereby declare that:

1. I am the inventor of the above-captioned patent application and the subject matter described and claimed therein.

2. Intel Corporation of Santa Clara, California, is the assignee of the above-captioned patent application.

3. I was employed by Intel Corporation at the time the above-captioned patent application was filed.

4. Prior to March 30, 1999, I conceived of the invention according to each of independent claims 1, 7, 11, 15, 19, 23, and 27 of the above-captioned patent application (hereinafter "the present invention") in this country, as evidenced by Exhibit A. The date stamp, in unredacted form, shows that the concept identified in the document was written prior to March 30, 1999.

5. Exhibit A shows a screenshot produced by an application program called Ecco. Ecco is a personal information manager (PIM) application. Among other uses, Ecco may be used to record notes, which are automatically date-stamped when they are entered. The screenshot shows a set of notes corresponding to a "Patents" tab, which relates to a PIM object I set up in Ecco to save information pertaining to patent concepts and the like – thus producing a type of electronic patent notebook. The screenshot shows information pertaining to conception of the present invention, including purpose, advantages, key elements, and steps used to perform the method.

6. I reduced the present invention to practice, with due diligence from a date prior to March 30, 1999 to March 22, 2000 (the filing date of the above-captioned patent application) in this country, as evidenced by Exhibit B.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the above-captioned application or any patent issued thereon.

Respectfully submitted,

Date August 4, 2004

Jay Connelly

Jay H. Connelly

patents:

resulting cache should be film1, film4 and film5. Films4 and 5 are then scheduled to be recieve
the cahce. Film1 and film3 are replaced.

- Return to step 2
- Others in this space
- Advantages to Intel

- method for determining and scheduling optimal broadcast content based on individual user preferences and ste
  - The purpose of this invention is to define a mechanism for scheduling broadcast content based on the agg users relevance information. It takes advantage of broadc ast economies of scale as well as personalizat the highest level of relevant content for a given user base.
  - Advantages of current systems
    - Much higher percentage of users influence the schedule
      - current broadcast systems select content in advance, often based on a wsubsampling of the entire population such as Neilson ratings.
    - maximizes revenue for broadcast pipe
    - minimizes end user wait time subject to maximizing revenue
    - scalable across millions of users.
    - Self-refining
  - Key elements
    - A client system to measure preferences
    - A client system to deliver measurements to service provider
    - A server system to aggregate preferences across the user base
    - A server system which broadcasts content based on the aggregate input from the user.
    - Can be combined with other relevance technology such as firefly
  - Steps
    - Initial broadcast content is sent out. The initial broadcast schedule is determined by legacy methods beyond the scope of this invention. For the case of video jukebox, this is likely based on historical data such as box office receipts.
    - A table of potential broadcast items is broadcast by the service provider and captured by each client system. For the case of video jukebox, this potential list of broadcast items also contains a set of meta data associated with each potential broadcast item since this is used in the rating algorithm.
    - Each client system rates the potential broadcast according to some mechanism. For example, this could be a combination of user input and inplicit inference as described in invention disclosure XYZ.
    - Each period (for example each night), each client sends updated feedback indicating the relevance of potential broadcast items.
    - An aggregate table is created at the broadcast origination site indicating aggregated relevance values for each potential broadcast.
    - This table is sorted according to user relevance
    - Assume the the system has bandwidth and time to broadcast N cache lines. The top N items are scheduled for broadcast.
    - In order to avoid stale relevance data, this table is destroyed after use and is re-created each day. That is, the relevance ratings are not aggregated over time.
  - Other intersections

Show Item Info

Item Text:

Folders and column values of item above:

Date Stamp	[redacted] Redacted

OK | Close | Prev | Next | Delete Item | Clear Value

Calendar | PhoneBook | Daily Journal | vjb | Completed Work | Broadband apps | 1:1 | RR | patents

8/2/04 10:47 AM

start | 6 Inter... | 2 Micro... | 2 Micro... | Rucker, ... | old eco's... | Ecco | 93% | 10:47 AM

P8387

INTEL INVENTION DISCLOSURE

LEGAL ID# 13532 DATE: 12/19/97 IAL COMM | IAL/BTL

P8899

It is important to provide accurate and detailed information on this form. The information will ~~be used~~ to evaluate your invention for possible filing as a patent application. When completed, please return this form to the **Legal Department at JF3-147**. If you have any questions, please call **264-0444 or 264-0998.**

1. Inventor: Connelly (Last Name) Jay (First Name) H (Middle Initial) 264-9684

SS# 542-84-1605 WWID 10068550 Phone ~~645-4481~~ M/S: jf2-86

Home Address: 3148 nw 126th place City Portland State or Zip 97229

Citizenship: US BUM Presenter: Carmen Egido

Group: (e.g. TMG, ICG, CEG) NBG Division Name IAL Subdivision BTL

Supervisor* Tony Baker WWID ____ Phone 264-8397 M/S: jf2-86

(PROVIDE SAME INFORMATION AS ABOVE FOR EACH ADDITIONAL INVENTOR)

2 Title of Invention: A signaling mechanism and apparatus to provide content on demand in a broadcast system.

3. What technology/product/process (code name) does it relate to: Digital Entertainment Initiative

4. Stage of development (i.e. % complete) 15

RECEIVED

JAN 04 2000

PATENT DATABASE GROUP

INTEL LEGAL TEAM

5. (a) Has a description of your invention been, or will it shortly be, published outside Intel:

NO: X YES: ____ DATE WAS OR WILL BE PUBLISHED: ____
If YES, was the manuscript submitted for pre-publication approval? YES: ____ NO: ____

(b) Has your invention been used/sold or planned to be used/sold by Intel or others?

NO: ____ YES: X DATE WAS OR WILL BE SOLD: Q100

(c) Does this invention relate to technology that is or will be covered by a SIG (special interest group)/standard/ or specification?

NO: X YES: ____ Name of SIG/Standard/Specification: ____

(d) If the invention is a semiconductor device, actual or anticipated date of tapeout? ____

(e) If the invention is software, actual or anticipated date of any beta tests. 2H/00

6. Was the invention conceived or constructed in collaboration with anyone other than an Intel blue badge employee or in performance of a project involving entities other than Intel, e.g. government, other companies, universities or consortia?

NO: X YES: ____ Name of individual or entity: ____

Please attach a page to this form, **DATED AND SIGNED BY AT LEAST ONE PERSON WHO IS NOT A NAMED INVENTOR**, to provide a description of the invention, and include the following information:

1 Describe in detail how the invention works

The purpose of this disclosure is to define a mechanism for providing content on demand in a broadcast system. This invention consists of a number of pieces. It is unclear which of the pieces are patentable in their own right so they are all presented here together (let the lawyers help sort them out). Specifically, the system consists of:

1. A mechanism for signaling and receiving meta data
2. A mechanism for using meta data in a broadcast system to establish relevance
3. A broadcast cache using a "consumable" policy.
4. A system which combines 1-3 in order to provide content on demand.
5. A mechanism to use the above to automatically select content for broadcast

These ideas were generated as part of the Video Jukebox investigation so we will use that example in the disclosure.

Given the movement we are seeing in the industry, we need to sew up some of this IP as soon as possible.

2 Mechanism for signalling and receiving meta data

In general, meta data can be considered as a set of descriptors that describe a set of content to be delivered over a broadcast pipe. These descriptors provide the ability for the client systems to think, reason, and make decisions about possible broadcast content to be delivered. Meta data forms the basis for Intel's client side filtering, cache management and other personalization techniques.

The signaling protocol involves sending proper signals such that the client systems can locate and acquire the broadcast content. This includes a new technique of pre-broadcast of the descriptors for upcoming content. Client systems capture and process this pre-broadcast information in order to determine when to receive content, where to receive content and which content to receive.

2.1 Essential Elements for signaling and receiving meta data

Scheduling service: Each client system contains a scheduling service which accepts requests to wake up at a specific time. This service causes the system to wake up at a the specified time and select a specific service. This selection process can be via tuning to a specific frequency (as in ATSC or a DVB transponder) or can be based on a set of data (such as multi-cast IP addresses) which defines a service. The exact scheduling service to be used is beyond the scope of this invention but is assumed to exist.

Broadcast signalling. The broadcast server must be capable of producing 4 signals:

1. A content descriptor pre-broadcast signal. A signal is emanated indicating when the content descriptors will arrive. This signal must contain information indicating when the descriptors will arrive as well as information indicating how to access the content.
2. A content descriptor broadcast signal. A signal containing the content descriptors. This is sent according to the specifics defined in the content descriptor pre-broadcast signal.
3. A content pre-broadcast signal. A signal indicating when the content will arrive along with selection information required to access the content.

4. A.content broadcast signal. This is a signal containing the actual broadcast content. It is sent according to and consistent with data contained in the content pre-broadcast signal.

Signal Bootstrap Service. Each client system maintains a signal bootstrap service that listens to a well known frequency/address in order to identify upcoming services. For the case of DVB and ATSC standards, this could be PID0 on a specific frequency. For the case of SAP, this could be the standard announcement port and IP address.

2.2 *Algorithm for signalling*

Figure 1 illustrates the signalling control flow diagram.

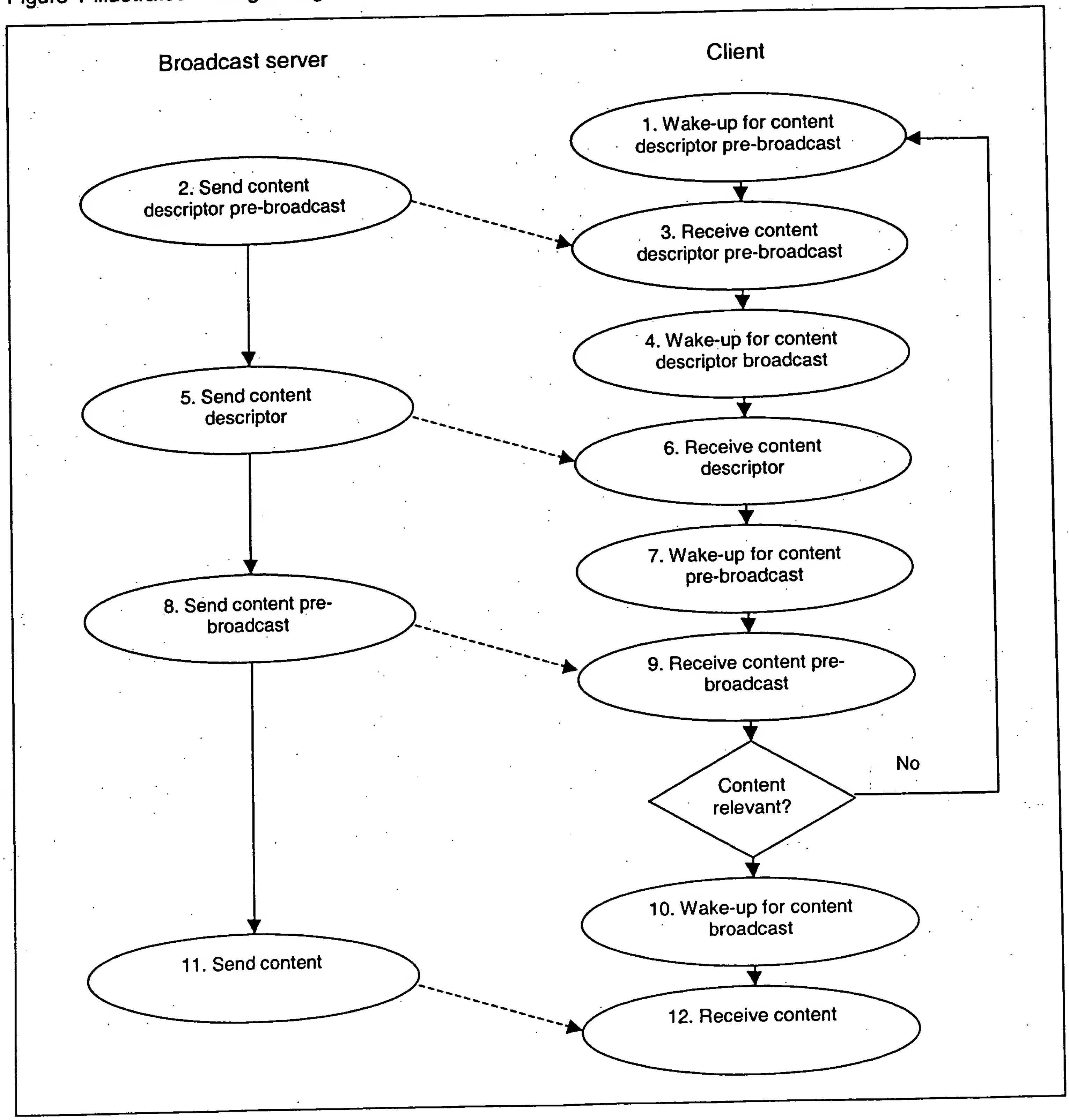

Figure 1: Signalling Control Flow Diagram

2.2.1 Client application registers to receive signals from content provider X

The client application registers with the client signaling system to receive signals from a specific content provider. The client signaling system maintains a table of applications associated with specific content providers.

For example, consider an application that registers to receive signals from the Intel_film service. The client signaling system maintains a table such as:

Content Provider	Application	Service
Intel_film_service	Intel_film_application	All services

Table tbd: broadcast signal mapping table

2.2.2 Content descriptor pre-broadcast is signalled

A signal is sent over the broadcast system indicating some point in the future when content descriptors (meta data) is going to be sent. The client system has been listening (could be a well known port such as used by PSIP, DVBSi or SAP) for the upcoming service announcement. The system schedules itself to wake up at that time and records details associated with the event.

For illustration purposes, assume that the following signal is formulated:

Content Provider	Event Description	Time	Frequency	Selection details
Intel_film_service	Content descriptor pre-broadcast	11/10, 0400	Channel 4	IP:A.B.C.D

The signal is sent over a well known address such that each client can use this well known address. For example, it is sent over channel 4, IP address: T.B.D. Assume that each client system has a service which is listening to this address.

2.2.3 Client system receives content descriptor pre-broadcast signal and wakeup is scheduled

When the content descriptor pre-broadcast signal arrives, the registered application is notified. In this example, the signalling system will wake up the Intel_film_application and will pass the pre-broadcast signal to the application. The application will in turn schedule itself to be revived at 0400 on 11/10.

2.2.4 The system wakes up and receives content descriptors

The system wakes up at the pre-specified time and receives the content descriptors. These descriptors are stored in the local content descriptor table. The meta data is then processed to decide which content to receive (see relevance and caching discussion for more detail). The application interested in the specific content is notified and can process the content descriptors to determine which content it is interested in.

2.2.5 Broadcast server sends content broadcast signal

The broadcast server sends a signal indicating when the actual content will arrive. Each client system interested in receiveing more content schedules itself to wake up when the content arrives.

For illustration purposes, assume that the following signal is formulated:

Content Provider	Event Description	Time	Frequency	Selection details
Intel_film_service	Content broadcast	11/10, 0600	Channel 4	IP:A.B.C.D

2.2.6 Client system receives content pre-broadcast signal and wakeup is scheduled

When the content broadcast signal arrives, the registered application is notified. In this example, the signaling system will wake up the Intel_film_application and will pass the broadcast signal to the application. The application will in turn schedule itself to be revived at 0600 on 11/10.

2.2.7 Content is broadcast to the system

The broadcast server then sends a set of content to all clients. Different clients will capture different content. The exact mechanism for selecting which content to capture is described in the section on relevance and cache management. The content is stored in the local cache for future, on-demand access.

3 Broadcast content on demand using relevance and a local cache

The purpose of the relevance and caching system is to allow clients in a broadcast system to maintain the most accurate cache of content for a specific domain. When combined with the signaling system (above), it allows each client to maintain its own set of content while allowing the broadcast system to only send out the actual content at specific times.

Note that although this disclosure describes one possible relevance algorithm, any algorithm capable of inferring relevance based on pre-broadcast of meta tags would work.

3.1 Essential Elements for content on demand in a broacast system

Each client system maintains a local <u>meta table</u>. The meta table might consists of 4 elements:

1. Meta data attribute. This is a general class for grouping like meta data
2. Meta data value. This is the specific value for an attribute. The attribute combined with the value constitues a combination key and the value/attribute key is the main concept used for rating.
3. Relevance Value. This is an indicator as to how relevant this attribute/value pair is for a given user. This is increased as the user expresses interest in a cache entry that has one of these attributes/value pairs in it and is decreased when the user indicates they do not want a cache item and the cache item has one of these attributes/value pairs in it.
4. Believability. This is a weight factor to be applied to the specific attribute/value pair. It is increased when an element accurately predicts elements a user is interested in and decreased when a user is interested in an element but the relevance indicated they would not be.

Entries in the meta table are constructed from the aggregation of all meta data associated with potential content and are updated based on explicit user requests as well as implicitly based on whether the user accesses the content or not.

<u>Content Rating Table</u>. A content rating table is maintained to maintain state information about which content is available as well as which content will be available at a later date. It also indicates how likely a user is to consume a piece of content (either currently cached or potential). Each tuple contains:

1. Name of the cache item. Used to describe the entry
2. Rating value for this item. This rating value is either manually input by the user or implicitly calculated by processing meta data associated with the cache line.
3. Rating treatment. This field either indicates 1) implicitly rate this line as things change or 2) explicit feedback was already provided, leave this rating alone.
4. Cache status to indicate if this item is currently cached locally.
5. Next treatment: used to track next actions to be taken for a given piece of content. In general, this is used to mark a currently cached item for deletion or a potential upcoming item for capture.

The local cache is the first level cache. Its size should be defined such that it optimizes the chance of a single hit.

Content Interaction signal. Each client system must be signalled when a user interacts with a piece of content. This can be done using something like a package delivery service. Each time the user interacts with a piece of content, the system is signalled and the relevance descriptors for that content are automatically updated in the meta table.

Content Descriptors. Each piece of content in the system must be formed to contain a set of value/attribute pairs describing the content. These descriptors are broadcast to each client. The client systems will capture the content descriptors and optionally present them to the consumer for rating. The value/attribute pairs are determined by the content provider and are input into the broadcast server.

Explicit feedback mechanism. Each client system must provide some mechanism for the users to express a preference to either receive the specified content or not to receive the specified content. Users are not required to provide explicit feedback although this option must be present for optimal behavior. This feedback mechanism is based soley on trimodal input. Either the user wants the content, they do not want the content or they express no preference. This mechanism may be a simple table indicating which content is available.

In order to demonstrate how these pieces fit together, a sample system operation is presented. For the purpose of this disclosure, the example system providing video on demand will be used. Assume that each cache line consists of a film. The meta data associated with each film is in the form of value attribute pairs. The attribute/value pairs are general and could consist of any number of attributes and any number of values.

3.2 Algorithm for broadcast content on demand cache management

3.2.1 Initial conditions

For the purpose of this disclosure, assume the system comes pre-loaded from the factory with three cache elements (films). The exact mechanism to bootstrap the content rating table is beyond the scope of this invention as the policy for maintaining the steady state system is much more important..

Content Description	Rating	Rating Type	Cached	Next treatment
Film1	0	NA	Yes	NA
Film2	0	NA	Yes	NA
Film3	0	NA	Yes	NA

Table TBD: Content rating table

At this time, we have no inofmration on how likely the user is to consume any specific cache element so the ratings are all zero.

3.2.2 Content provider assigns meta data to broadcast content

The content provider has a set of tools which allow them to associate meta data with upcoming broadcast content and which allow them to produce the required signaling for the system. For the purpose of this disclosure, assume that the domain is films and the meta data consists of canonical descriptions of each film such as actors and genre. More complex systems can easily be built using the existing mechanisms..

For example, consider the following broadcast content and associated meta data. Each film is modeled as a potential cache line.

Film_description	Attribute	Value
Film1	Actor	Actor 1
	Genre	Action

Table TBD: Sample entry description table for film1

Film_description	Attribute	Value
Film2	Actor	Actor 2
	Genre	Comedy

Table 2: Sample entry description table for film2

Film_description	Attribute	Value
Film3	Actor	Actor 2
	Genre	Action

Table 2: Sample entry description table for film3

Film_description	Attribute	Value
Film4	Actor	Actor 1
	Genre	Comedy

Table 2: Sample entry description table for film4

Note that for this example, the film_descriptor is simply a name but in real life, the film would likely be represented by a more complex structure such as name, description and possibly a preview video.

3.2.3 Client receives content descriptors, creates content rating table and meta table

Assume the client system receives the content descriptors based on the signaling technique described in section 2.

The content rating table is constructed using all items that may be broadcast. For the case of the film service, assume that there are a number of films that may be sent. The rating value represents how likely the user will be to watch the film. Initially, all ratings are zero:

Content Description	Rating	Rating Type	Cached	Next treatment
Film1	0	NA	Yes	NA
Film2	0	NA	Yes	NA
Film3	0	NA	Yes	NA
Film4	0	NA	No	NA

The meta table is contructed with all the attributes present for any of the films. For the running example, the following meta table is constructed:

Attribute	Value	relevance	Believability
Actor	Actor 1	NA	NA
Actor	Actor 2	NA	NA
Genre	Action	NA	NA
Genre	comedy, ...	NA	NA

Table TBD: initial meta data table

No relevance or believability information is known at this time.

3.2.4 The user optionally rates the potential content

The user is optionally presented the ability to request or dismiss potential content. This is not a required step as the caching algorithm will work without any explicit user ratings. By allowing this to be an optional step, this system promotes scalable interactivity. This means that if users want to interact with

the system, they can fine tune to behavior but for consumers who do not wish any more "technical interaction", the system can still function.

For the illustration example, assume the following is presented to the user:

Content Description	Treatment (receive/refuse)
Film1	Receive (user requested)
Film2	Refuse (user requested)
Film3	No action specified
Film4	No action specified

Table TBD: Content explicit rating table

Assume that the user rates item1 and item2. They request to receive film1 and not to receive film2. No preference is made for films 3 and 4.

3.2.5 Step7: Update the meta table based on ratings

For each item that the user requested, the relevance value for all attributes associated with that content is incremented. For each item that is dismissed by the user, the relevance value for all attributes associated with the content is decreased.

For the running example, the following meta table is constructed based on the combination of user input and content descriptors.

Attribute	Value	relevance	Believability
Actor	Actor 1	1	0
Actor	Actor 2	-1	0
Genre	Action	1	0
Genre	comedy, ...	-1	0

Table TBD: meta data table

Note that the relevance rating for each attribute of filjm1 is incremented (they started at zero) to indicate that all data to date indicates that the user is interested in this item. Also note that the relevance values for film2 are decremented to indicate that data so far suggests the user is not interested in films with these qualities.

Also note that since the user has not actually interacted with any of the content, the believability associated with each tuple is not known.

3.2.6 User consumes one of the cache elements

An important aspect of this system is that the local cache is modelled as a single use (fire and forget) system. This means that when a user accesses a cache line, they are not likely to want to acces that same cache line again.

Assume that the user watches film1. The relevance system now needs to be updated to reflect that fact.

3.2.7 Update meta table based on interaction

Each time the user interacts with the content, the believability of the value attribute pairs for each descriptor in that film is updated. For value/attribute pairs which have a relevance value greater than zero, the believability factor for that attribute is increased (since it accurately served as a predictor for which pieces a user would interact with). For attributes which have a relevance value less than zero, the believability value for this attribute is decreased (since it did not accurately predict which which items the user would interact with)..

For the running example, the meta table now looks like:

Attribute	Value	relevance	Believability
Actor	Actor 1	1	1
Actor	Actor 2	-1	0
Genre	Action	1	1
Genre	comedy, ...	-1	0

Table TBD: meta data table

Note that the relevence field is not updated for films that have been implicitly related. This must be tracked (in the content table) such that the relevance values can make ~~progress~~. If the user ever interacts with a piece of implicitly rated content, rhe relevance values must be adjusted along with the believability attributes.

3.2.8 Update content rating table

Now that the system has some actual data on what the user will interact with, it needs to go back and update the ratings for the existing content. The ratings are based on:

For each attribute: ~~relevance~~ component rating value = attribute_relevance*believability

For the content: cache rating = average of all attribute relevance values with positive believability

For example, the rating for film3 is:

Film	Attribute	Value	Relvance from meta table	Believability from meta table	Component rating value
Film3	Actor	Actor 2	-1	0	0
	Genre	Action	1	1	1
Total					0.5 (average of above)

Table 2: Temporary cache rating table for film3

The new content table looks like:

Content Description	Rating	Rating Type	Cache Treatment	Next treatment
Film1	1	Explicit	In cache	Replace
Film2	0	Explicit	In cache	Replace
Film3	.5	Implicit	In cache	~~Capture~~ real
Film4	.5	Implicit		Capture

Table TBD: Content rating table

3.2.9 Step 14: The user interacts more with the system

All future interaction results in similar treatment. For example, consider that the user watches film3. This is an interesting corner case since film3 has not been rated classified and the relevance values for each attribute have not been updated in the meta table. In this instance, both the relevance and the believability items must be updated.

Attribute	Value	relevance	Believability
Actor	Actor 1	1	1
Actor	Actor 2	-1	-1
Genre	Action	2	2
Genre	comedy, ...	-1	0

Table TBD: meta data table

Note that since the user watched film3 and since film3 was an action movie, the system is starting to rate action movies higher. Since the user dismissed film2 (which has actor 2) but then watched film3 (which had actor2), the system is now indicating that actor 2 is not a predictor of user behavior.

3.2.10 Maintaining the most accurate cache lines

Each time the user interacts with the system, the content table and the meta table are updated. This accounts for cases where the user consumes a piece of content as well as cases where they rate a new piece of content. By using the meta data describing the upcoming content to be broadcast combined with the content rating table (and algorithm), the system can accurately select and acquire the most appropriate content.

4 Advantages over existing systems

Using a local broadcast cache allows content providers to manage their network efficiently. This invention solves the scalability problems associated with all known existing content on demand systems.

Scales across millions. This system will work for a broadcast system with one client and for a broadcast system with billions of clients. The incremental cost to a service provider for each new user is zero. The same content is broadcast (independent of the number of users) as are the same content descriptors. Conversely, in a system which requires point to point communication (such as the web or more traditional content on-demand systems) also requires content providers to send content to each user when requested. Also, a system which maintains relevance on the server requires incremental resources for each new user.

Broadcasting the potential cache list ahead of time allows client systems to prepare for new cache lines when they arrive. In a broadcast system, it is possible (and likely) that there are a number of potential things to download. By pre-broadcasting all the possible entries ahead of time, the client system can determine which resources to collect and when.

Allows relevant content to be played directly from the hard drive rather than streamed over the network, which provides better responsiveness and control.

Rating all potential broadcast items can be used to provide valuable feedback to the service provider. See invention disclosure titled: "Driving a broadcast schedule through independent rating systems."

As long as at least on cache element is attractive to the user, this system provides content on demand. In order to be most valuable, the time to consume multiple pieces of content should be greater than the time the system needs to provide new content. For example, if we assume that films are broadcast every night and we assume that users only watch 1 film per night, the system can maintain a highly relevant cache.

By broadcasting the entire pre-broadcast cache, users can see all the content they might possibly get. This is in stark contrast to existing video broadcast systems where the users have no idea which content the broadcaster has.

Users are not required to interact with the rating system in order to accurately predict relevance. This system can work on 100% implicit relevance (based soley on which content the user interacts with) or can be based on a combination of implicit relevance combined with explicit user feedback. Most of the existing relevence systems require input from the user to improve over time. This system infers relevance based on the content that the user actually interacts with.